

U

O

W

Deploying the New System

CSIT114 / 814: Systems Analysis



UNIVERSITY
OF WOLLONGONG
AUSTRALIA

Reading

- This week: Chapter 14

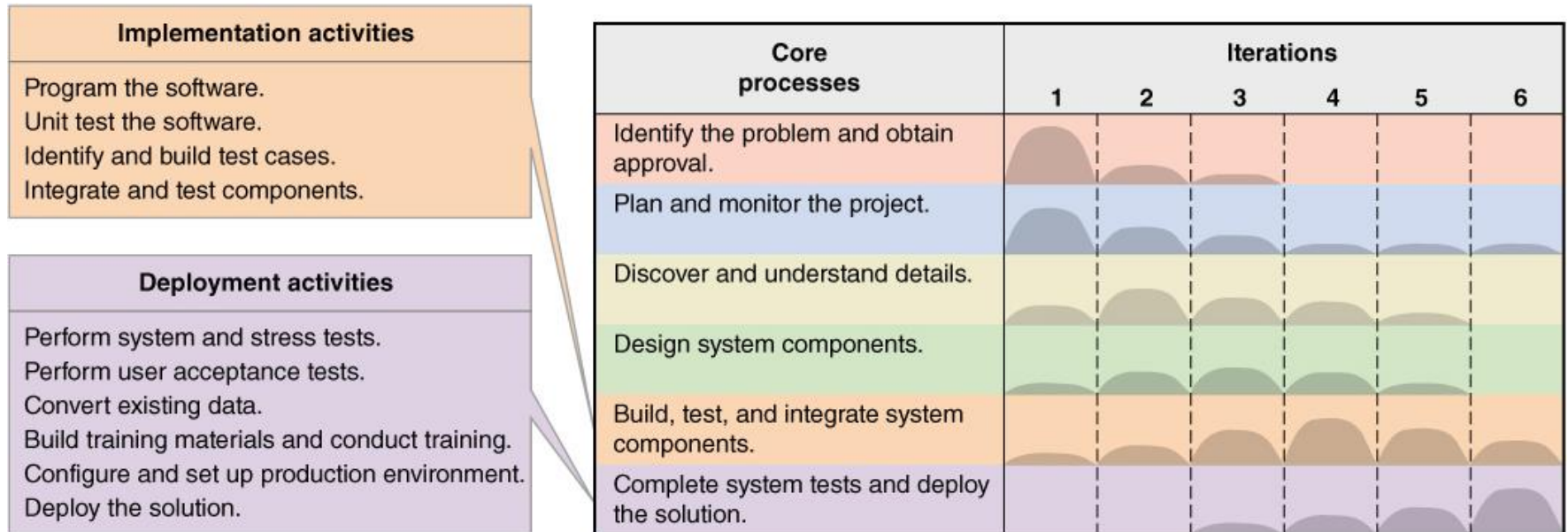
Outline

Testing

Deployment Activities

Planning and Management

Core Processes with Implementation and Deployment in SDLC



Testing

- Why do we need to test system before the deployment?
- What are the essentials for the system testing?
- What are the different types of testing?

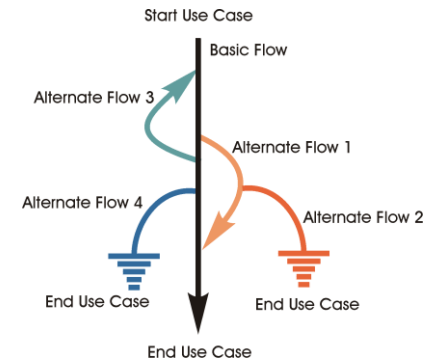


Testing Concepts

- Testing – the process of examining a component, subsystem, or system to determine its operational characteristics and whether it contains any defects.

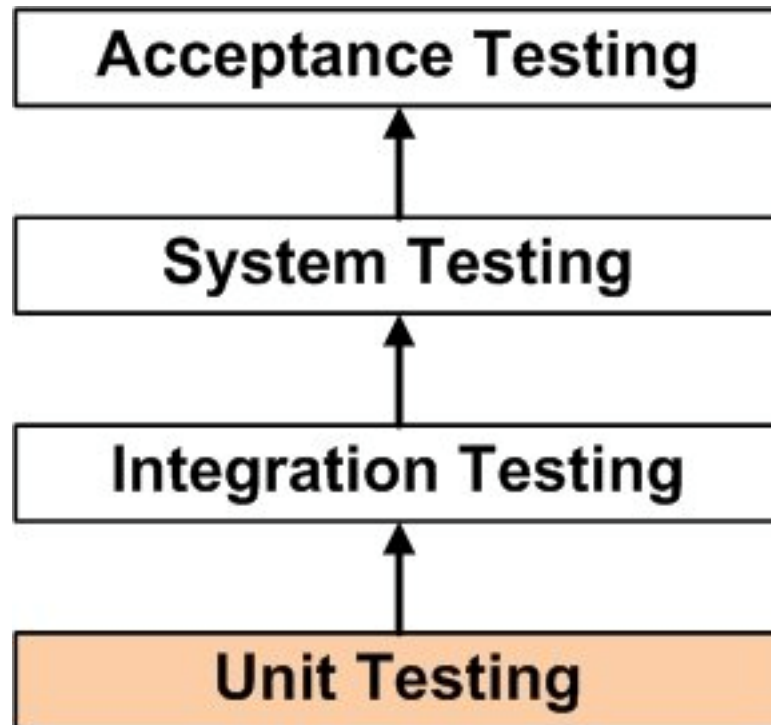
Testing Concepts

- Test case – a formal description of a starting state, one or more events to which the software must respond, and the expected response or ending state
 - Defined based on well understood functional and non-functional requirements
 - Must test all normal and exception situations
- Test data – a set of starting states and events used to test a module, group of modules, or entire system
 - The data that will be used for a test case



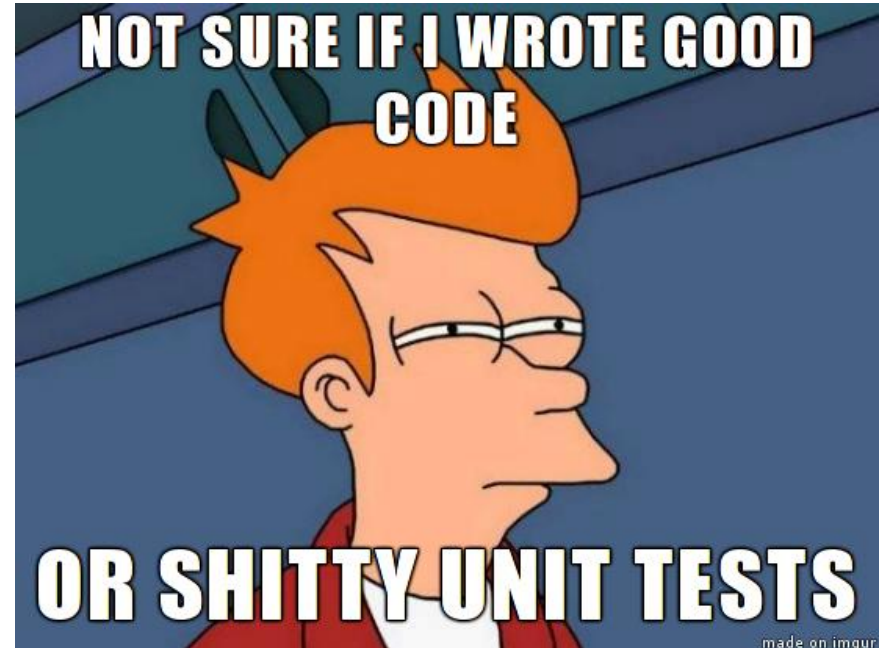
Most common types of tests

Test type	Core process	Need and purpose
Unit testing	Implementation	Software components must perform to the defined requirements and specifications when tested in isolation—for example, a component that incorrectly calculates sales tax amounts in different locations is unacceptable.
Integration testing	Implementation	Software components that perform correctly in isolation must also perform correctly when executed in combination with other components. They must communicate correctly with other components in the system. For example a sales tax component that calculates incorrectly when receiving money amounts in foreign currencies is unacceptable .
System and stress testing	Deployment	A system or subsystem must meet both functional and non-functional requirements. For example an item lookup function in a Sales subsystems retrieves data within 2 seconds when running in isolation, but requires 30 seconds when running within the complete system with a live database.
User acceptance testing	Deployment	Software must not only operate correctly, but must also satisfy the business need and meet all user “ease of use” and “completeness” requirements—for example, a commission system that fails to handle special promotions or a data-entry function with a poorly designed sequence of forms is unacceptable.



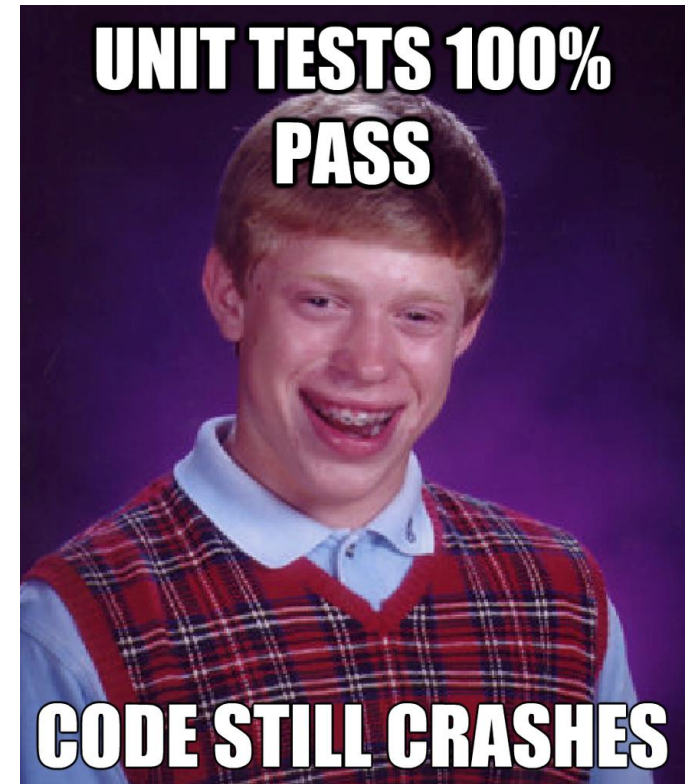
Unit Testing

- The test data and the test are done by programmer.
 - The fastest and easiest approach to unit testing.
 - The responsibility of writing solid, clean code right on programmer.



Integration Testing

- An integration test evaluates the behavior of a group of methods, classes, or components.
- The purpose of integration test is to identify errors that weren't or couldn't be detected by unit testing.



Integration Testing

- The complexity of integration test increases as the system grows. If multiple programmers are involved, several procedures must be put in place.



System, Performance, and Stress Testing

- System test – an integration test of an entire system or independent subsystem.
 - Can be performed at the end of each iteration
 - Can be performed more frequently

System, Performance, and Stress Testing

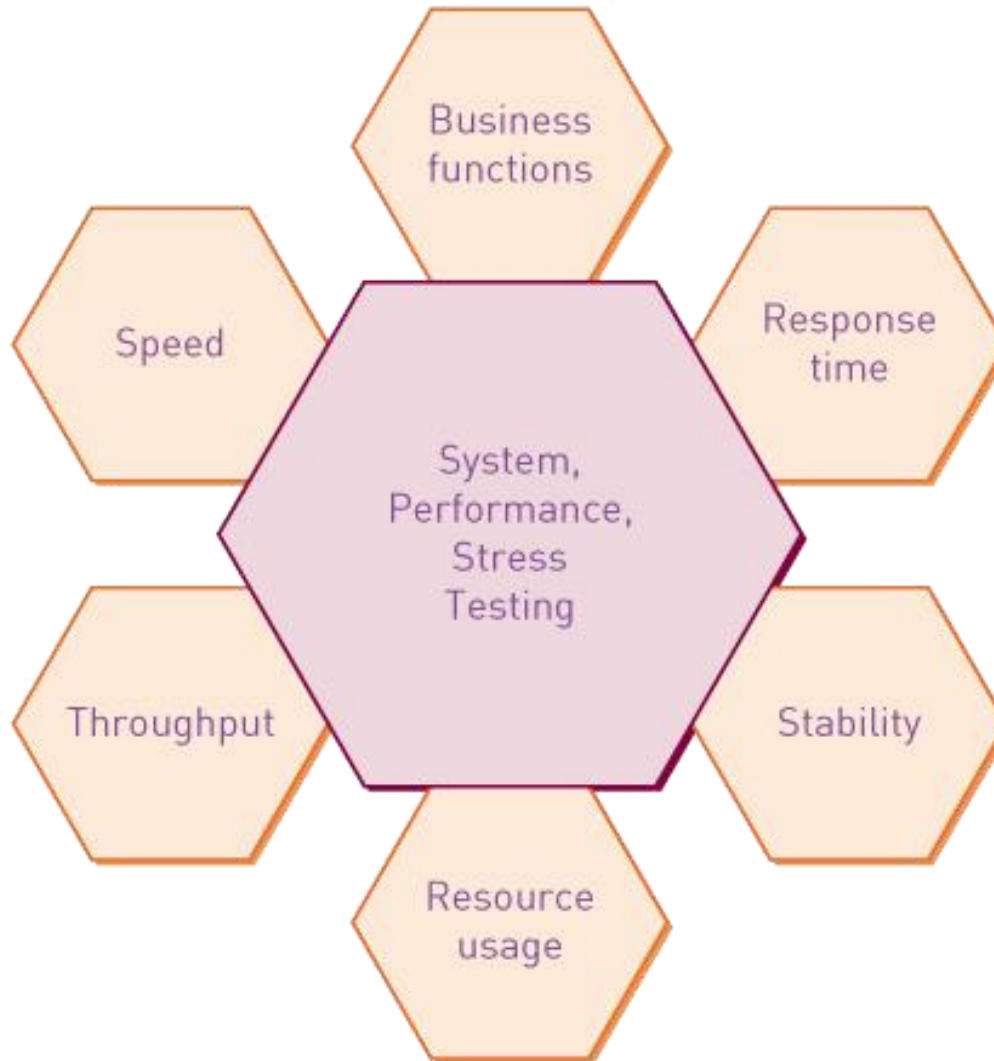
- Build and smoke test – a system test that is performed daily or several times a week
 - The system is completely compiled and linked (built), and a battery of tests is executed to see whether anything malfunctions in an obvious way (“smokes”)
 - Automated testing tools are used. Catches any problems that may have come up since the last system test



System, Performance, and Stress Testing

- Performance test or stress test – an integration and usability test that determines whether a system or subsystem can meet time-based performance criteria
 - Response time – the desired or maximum allowable time limit for software response to a query or update
 - Throughput – the desired or minimum number of queries and transactions that must be processed per minute or hour

System, Performance, and Stress Testing

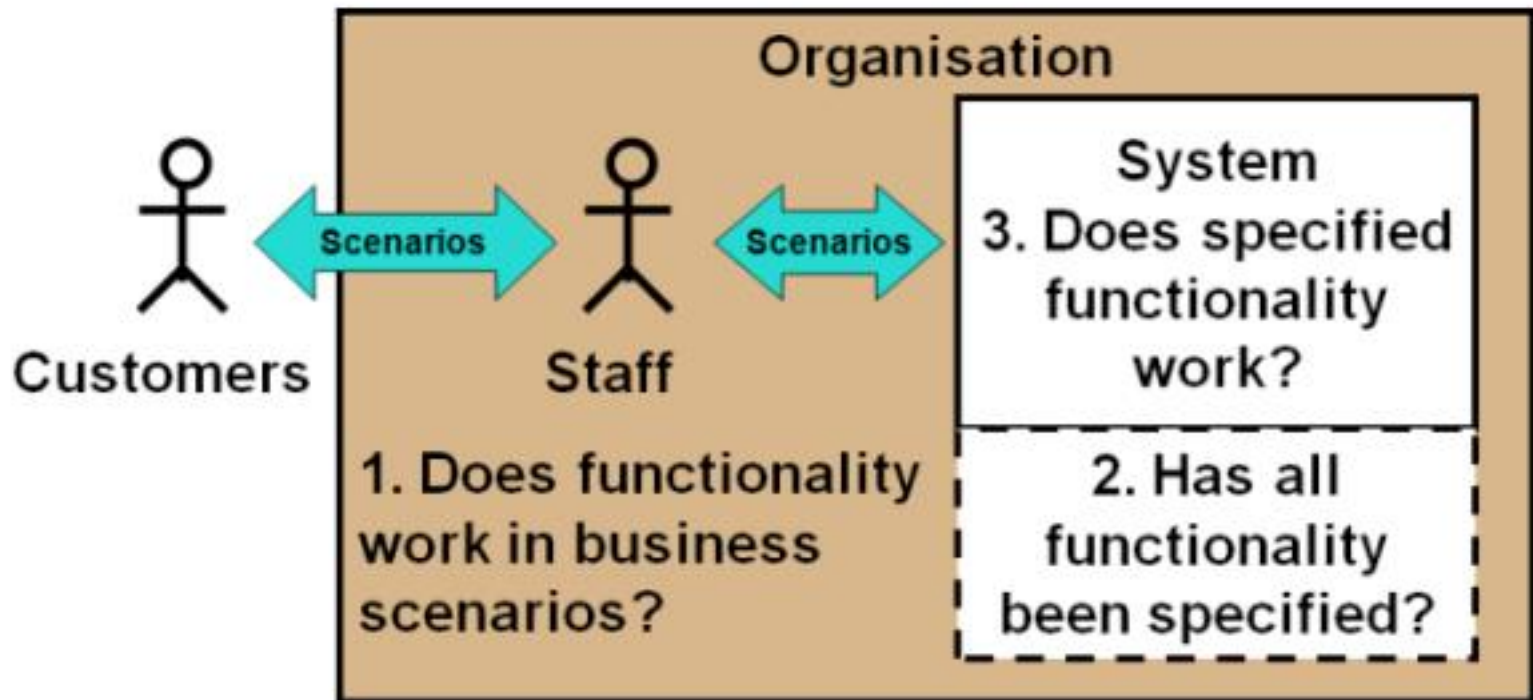


User Acceptance Testing (UAT)

- A user acceptance test is a system test to determine whether the system fulfills user requirements.
- The UAT is normally the final stage in testing the system.
 - A very formal activity in most development projects. Payments tied to passing tests
 - Details of acceptance tests are sometimes included in the request for proposal (RFP) and procurement contract.

UAT Objectives

Outside World



Copyright Coley Consulting 2001-2011

User Acceptance Testing (UAT)

- Plan the UAT
 - Should be done early in the project
 - Test cases for every use case and user stories
 - Identify conditions to verify that the system supports the use case accurately and completely
- Sample test case list

	A	B	C	D	E	F
1	Spec ID	Cross refer to use case	Short description	Test conditions	Expected outcomes	Comments
2	10	101	Maintain customer Info	Add customer, update customer, delete not allowed	New customer with all fields, updated customer with selected fields	
3	11	201	Maintain sale info	Create sale, update sale, finalize sale, pay for sale	New sale in DB, update selected fields, payment creates transaction	
4	12	202	Ship items	Display items, update status	Sale update, sale items updated, shipment created	



User Acceptance Testing (UAT)

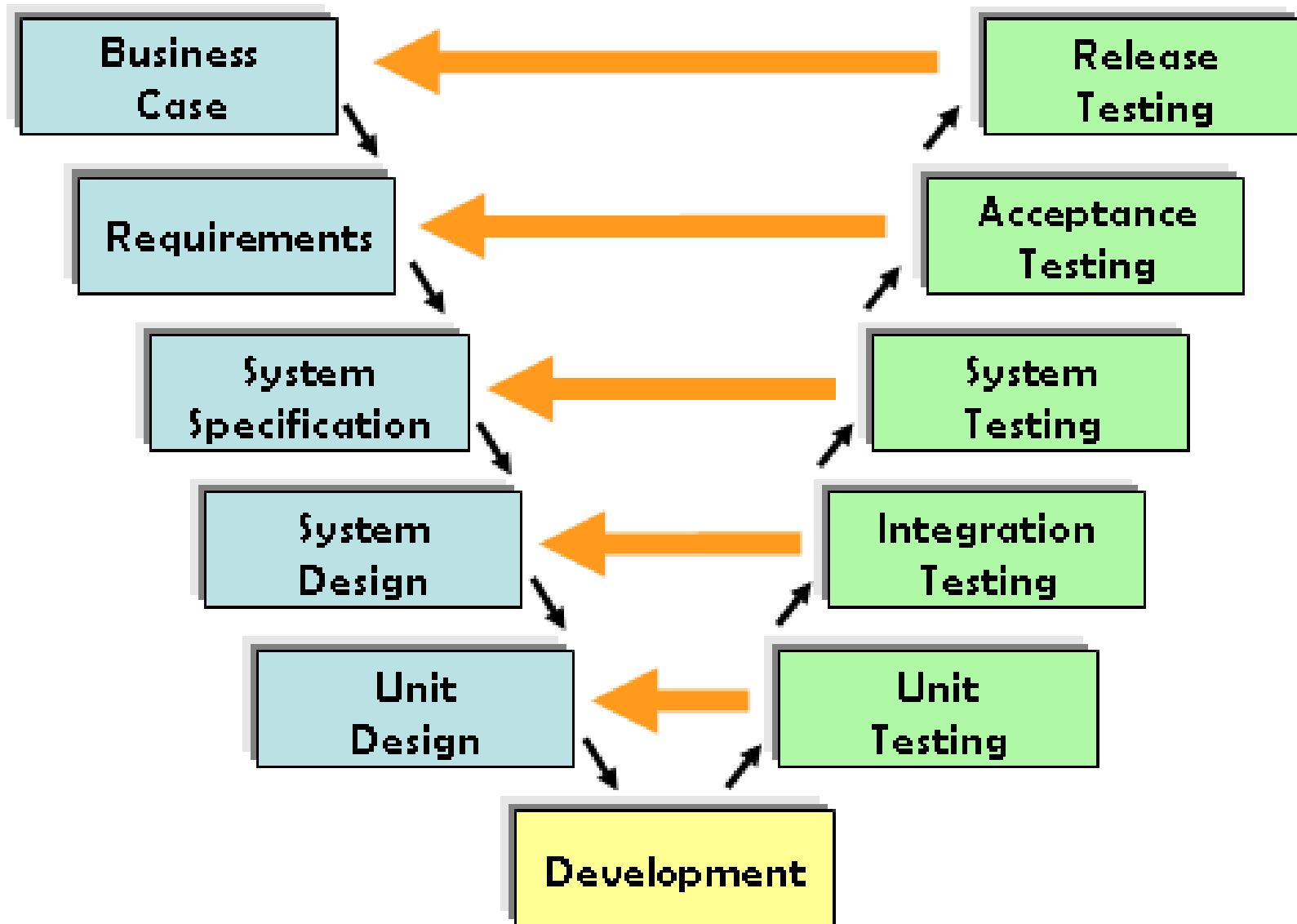
- Preparation and Pre-UAT Activities
 - Develop test data – data entry and database records
 - Plan and schedule specific tests
 - Set up test environment

Acceptance Tests



User Acceptance Testing (UAT)

- Manage and execute the UAT
 - Much like a mini-project
 - Assign responsibilities
 - Document and track results (especially errors and fixes)
 - Rework the plan for re-testing as required



Quick quiz

- What is the difference between integration testing and system testing?
- A: Integration testing is done on a single subsystem and focuses on the communication between components. System testing is done on the complete system, or multiple subsystems and focuses on not just integration but other issues such as performance.

Quick quiz

- What is the difference between system testing and user acceptance testing?
- A: System testing does include the entire system, but focuses more on correctness of results. User acceptance testing does include correctness of results, but also includes usability testing and actually solving the business need.

Deployment activities

Converting and Initializing Data

- An operational system requires a fully populated database to support ongoing processing
- Data needed at system startup can be obtained from these sources:
 - Files or databases of a system being replaced
 - Manual records
 - Files or databases from other systems in the organization
 - User feedback during normal system operation

Converting and Initializing Data

- Reuse existing databases
 - Modify or update existing data
 - Supported by most DBMSs
- Reload databases
 - Copy and convert the data
 - Export and import data from distinct DBMSs
 - Built-in and third-party utility programs, or customized programs
 - Data entry from paper documents
 - “Import on use”

Training Users

- Training is needed for end users and system operators



Training Users

- End users:
 - General staff who use the system from day to day to achieve the system's business purposes.
- System operator:
 - Technical staff who perform administrative functions and routine maintenance to keep the system operating.

Training Users

End-user activities	System operator activities
Creating records or transactions	Starting or stopping the system
Modifying database contents	Querying system status
Generating reports	Backing up data to archive
Querying database	Recovering data from archive
Importing or exporting data	Installing or upgrading software

Training Users

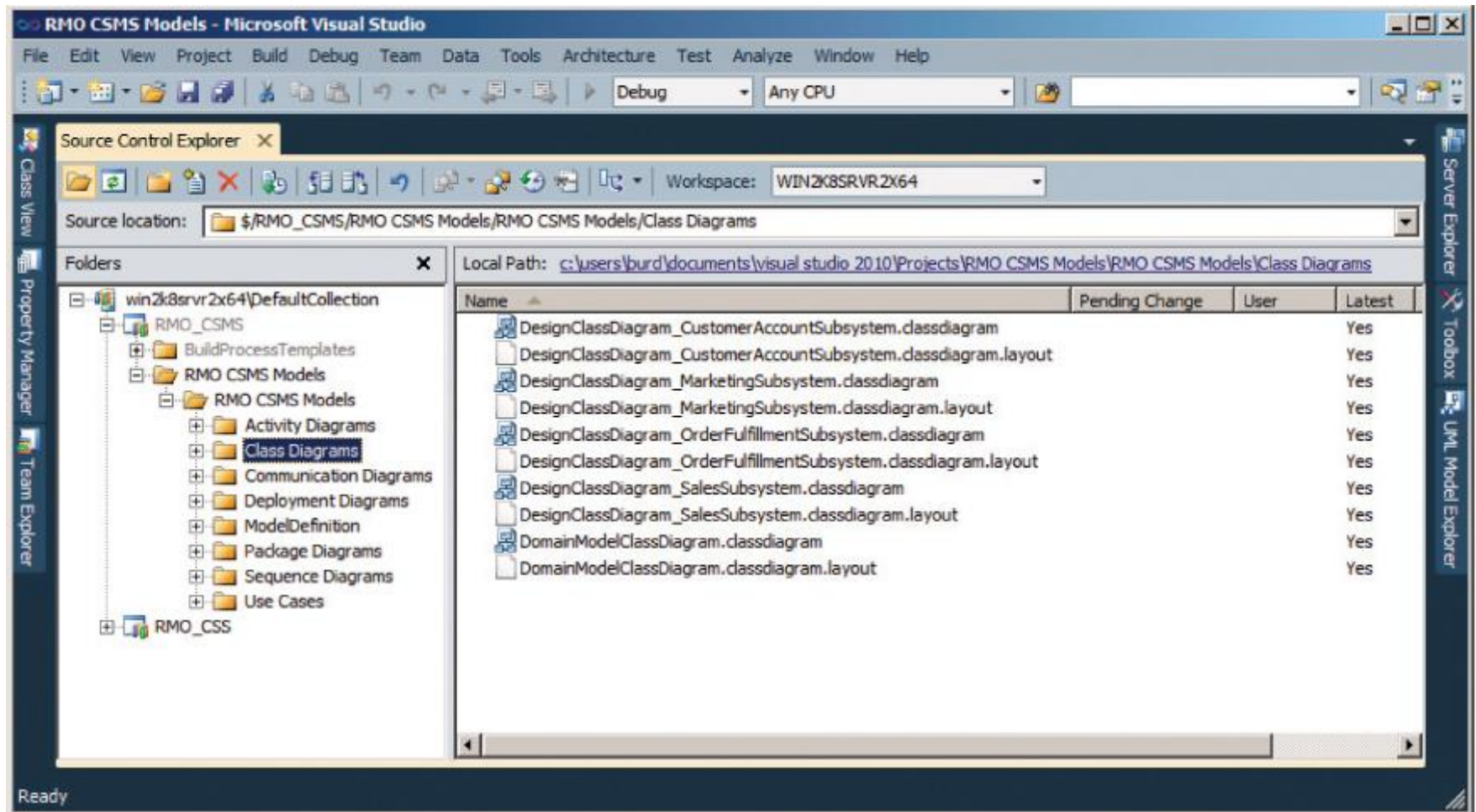
- Training for end users must emphasize hands-on use for specific business processes or functions, such as order entry, inventory control, or accounting
 - Widely varying skill and experience levels call for at least some hands-on training, including practice exercises, questions and answers, and one-on-one tutorials
- System operator training can be much less formal when the operators aren't end users
 - Experienced computer operators and administrators can learn most or all they need to know by self-study

Training Users

- System Documentation
 - Descriptions of system requirements and architecture to help maintenance and upgrade of the system
- User Documentation
 - How to interact with and use the system for end users and system operators

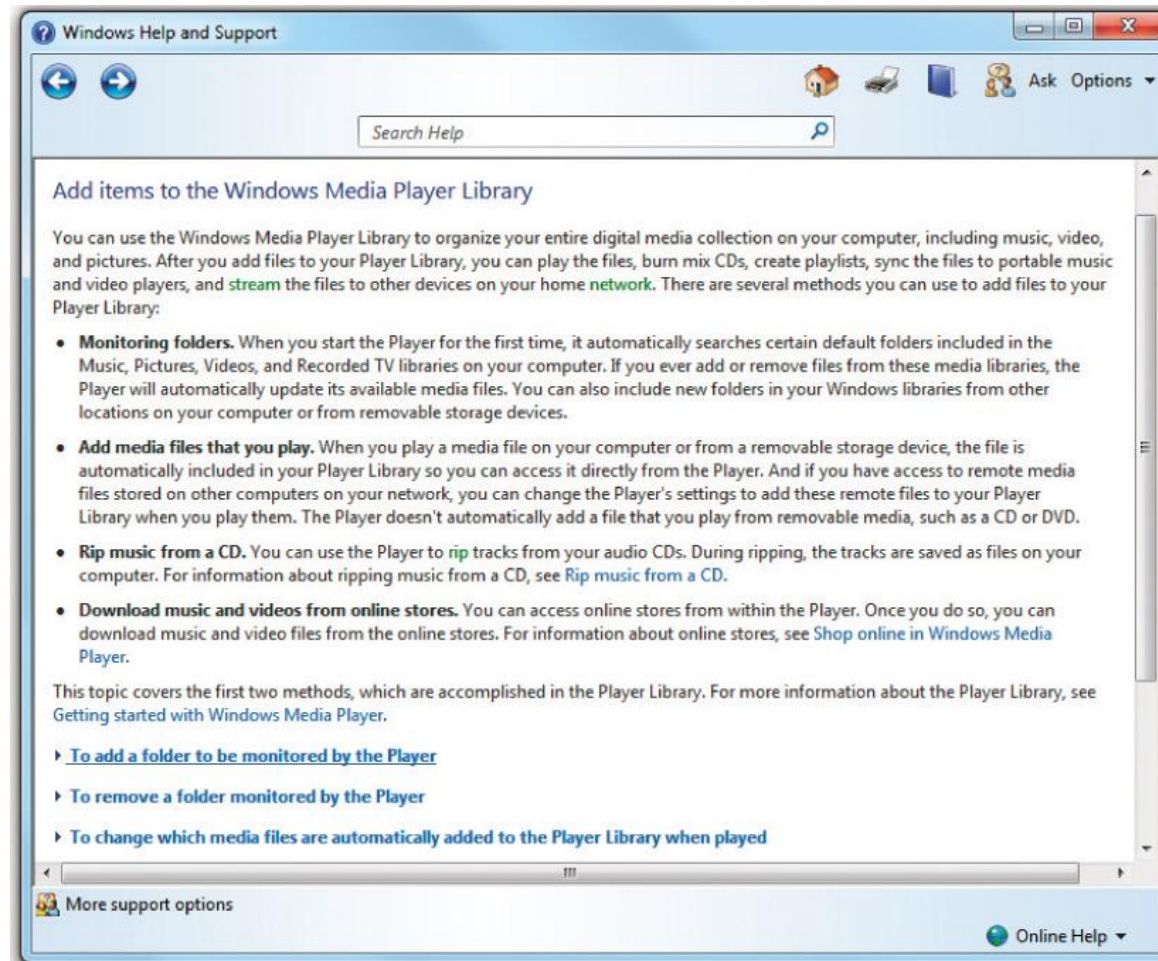
Training Users

- System Documentation



Training Users

- User Documentation



Managing implementation

Planning and Managing

Implementation, Testing and Deployment

- Development Order

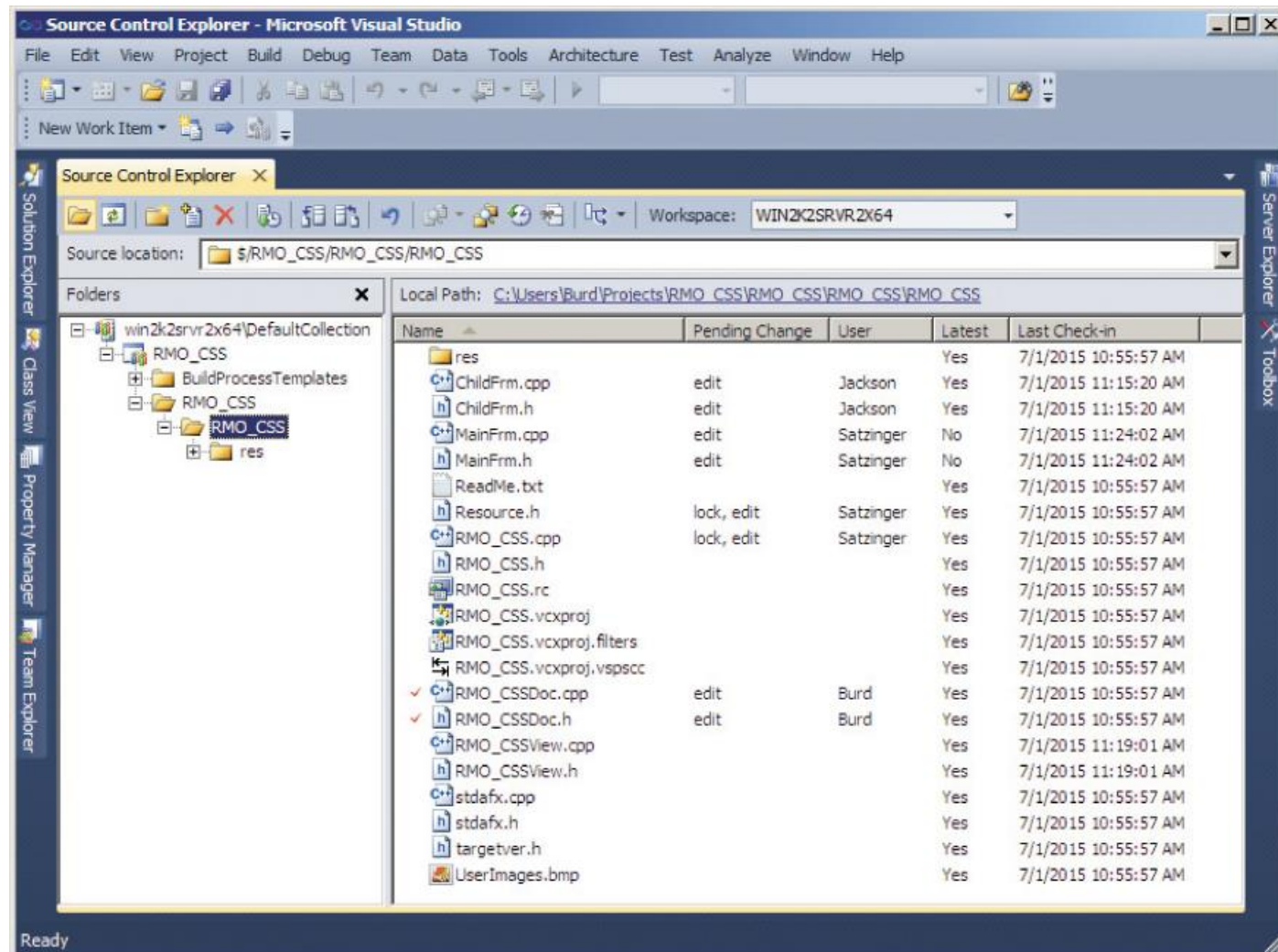
- Input, process, output (IPO) – a development order that implements input modules first, process modules next, and output modules last
- Top-down development – a development order that implements top-level modules first
 - Use stubs for testing
- Bottom-up development – a development order that implements low-level detailed modules first
 - Use drivers for testing
- Use-case driven – select specific use cases and order the development based on selected use cases

Planning and Managing

Implementation, Testing and Deployment

- Source code control
 - An automated tool for tracking source code files and controlling changes to those files
 - A programmer checks out a file in read-only mode when he or she wants to examine the code without making changes (e.g., to examine a module's interfaces to other modules)
 - When a programmer needs to make changes to a file, he or she checks out the file in read/write mode
 - The SCCS allows only one programmer at a time to check out a file in read/write mode.

Source Code Control System (SCCS)



Planning and Managing

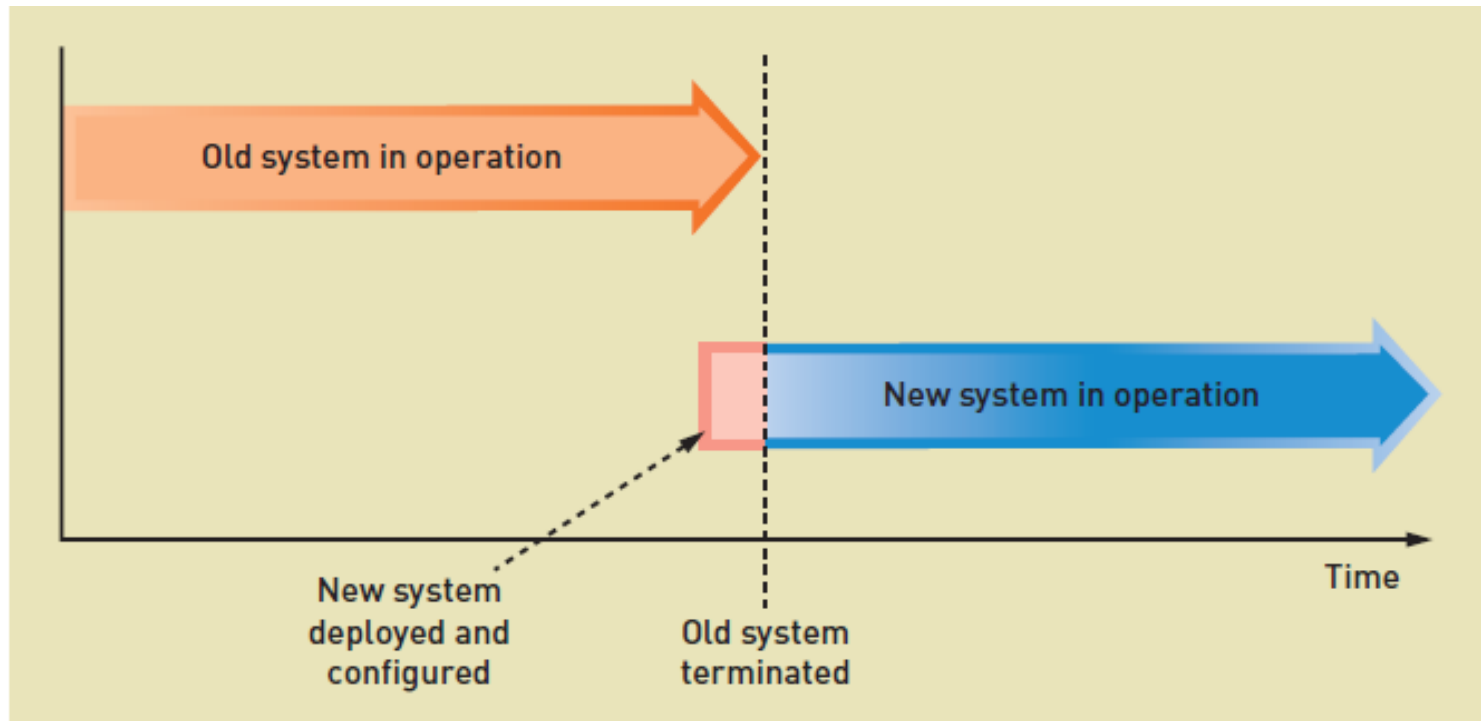
Implementation, Testing and Deployment

- Packaging, installing, and deploying components
 - Issues to consider when planning
 - Incurring costs of operating both systems in parallel
 - Detecting and correcting errors in the new system
 - Potentially disrupting the company and its IS operations
 - Training personnel and familiarizing customers with new procedures
 - Different approaches
 - Direct deployment
 - Parallel deployment
 - Phased deployment

Planning and Managing

Implementation, Testing and Deployment

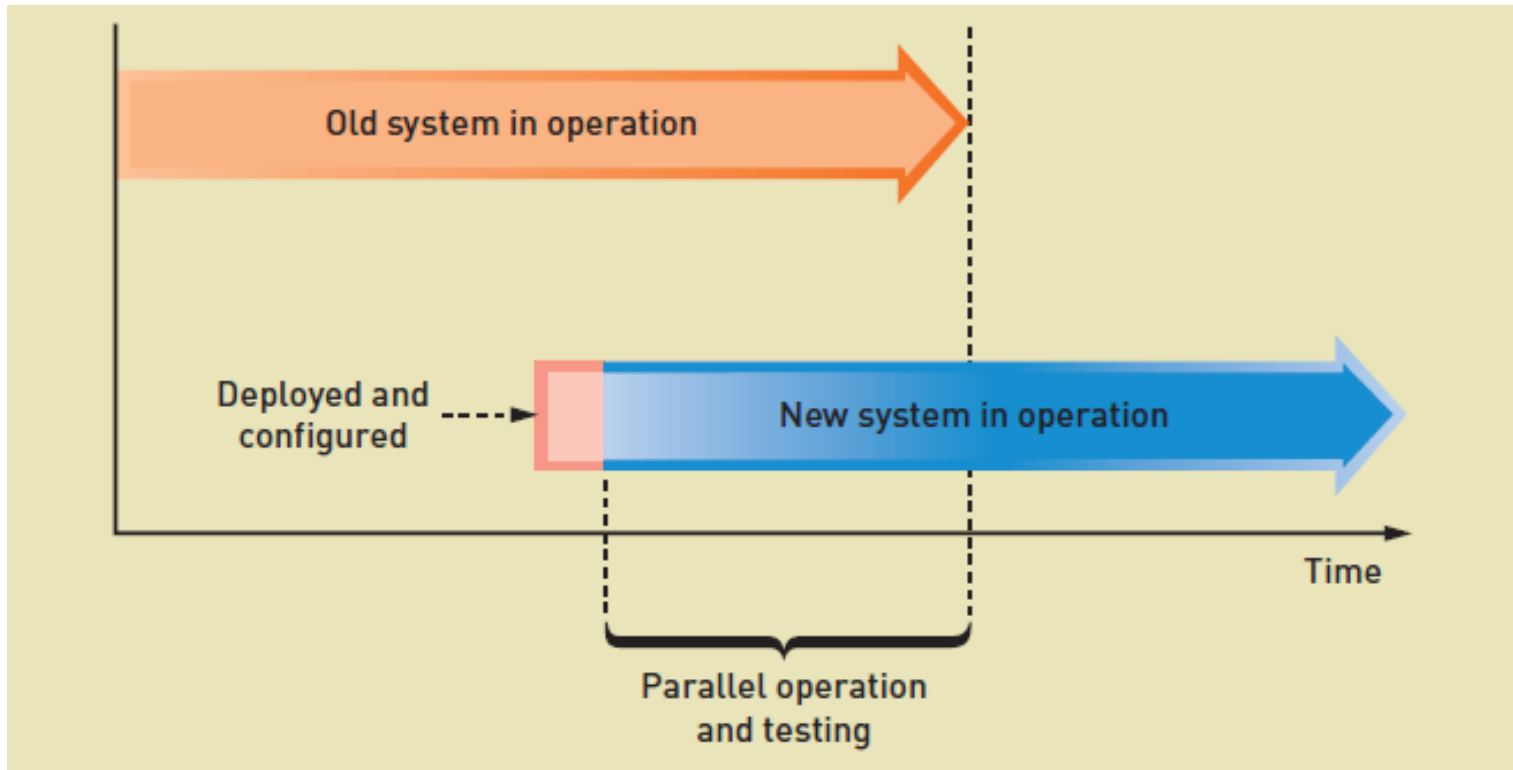
- Direct deployment – a deployment method that installs a new system, quickly makes it operational, and immediately turns off any overlapping systems
 - Higher risk, lower cost



Planning and Managing

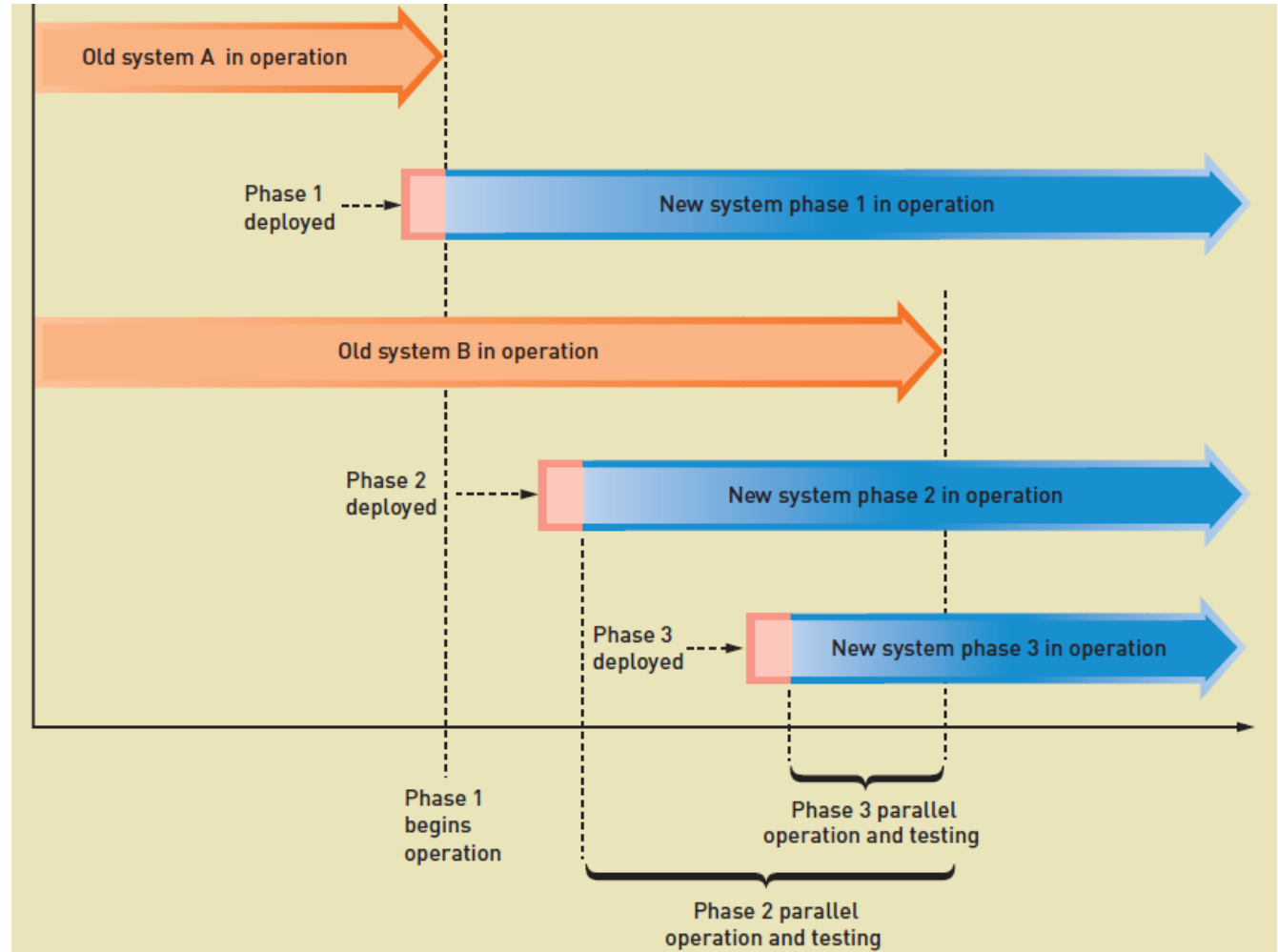
Implementation, Testing and Deployment

- Parallel deployment – a deployment method that operates the old and the new systems for an extended time period
 - Lower risk, higher cost



Planning and Managing Implementation, Testing and Deployment

- Phased deployment –
- a deployment method that installs a new system and makes it operational in a series of steps or phases



Support Activities

- The objective of support activities is to keep the system running productively during the years after it is installed and in use.
- Major support activities:
 - Maintaining the system
 - Enhancing the system
 - Supporting the users

Planning and Managing

Implementation, Testing and Deployment

- Change and Version Control – tools and processes handle the complexity associated with testing and supporting a system through multiple versions
 - Alpha version – a test version that is incomplete but ready for some level of rigorous integration or usability testing
 - Beta version – a test version that is stable enough to be tested by end users over an extended period of time
 - Production version, release version, or production release – a system version that is formally distributed to users or made operational for long-term use
 - Maintenance release – a system update that provides bug fixes and small changes to existing features

Planning and Managing

Implementation, Testing and Deployment

- Submitting Error Reports and Change Requests
 - Standard reporting methods
 - Review of requests by a project manager or change control committee
 - For operational systems, extensive planning for design and implementation
- Implementing a Change
 - Identify what parts of the system must be changed
 - Secure resources (such as personnel) to implement the change
 - Schedule design and implementation activities
 - Develop test criteria and a testing plan for the changed system

Quick quiz

- What does IPO stand for and what does IPO development order mean?
- A: IPO = Input, process, output. It means the order of developing the system is to develop the portions of the system that accept input first, then those parts that do the processing, then those parts that produce outputs.
- What is the difference between parallel deployment and phased deployment?
- A: Parallel is when the old system and the new system are both being used at the same time. Phase deployment is when parts of the new system are put into production. It is deployed a subsystem at a time.

Summary

- Implementation and deployment are complex processes because they consist of so many interdependent activities
- Implementation activities include program the software, unit tests, building test cases, and integrate and test components
- Deployment activities include perform system and stress tests, perform acceptance tests, convert existing data, build training materials/conduct training, configure and set up the production environment, and deploy the solution
- Testing is a key activity of implementation and deployment and includes unit tests, integration tests, usability tests, system/performance/stress tests, and acceptance tests

Summary (continued)

- Three options for deployment include direct deployment, parallel deployment and phased deployment
- Direct deployment is riskier but less expensive. Parallel deployment is less risky but more expensive
- For moderate to large projects, a phase deployment approach makes sense to get key parts of the system operational earlier

U

O

W

Questions



UNIVERSITY
OF WOLLONGONG
AUSTRALIA